Scattered Data Near-Interpolation with Application to Discontinuous Surfaces

Renata Besenghi and Giampietro Allasia

Abstract. This paper discusses a particular type of function approximation on scattered data in a general number of variables, and its application to surface representation with imposed conditions. If the given function values are subject to errors, it is not appropriate to interpolate the function at the data in the sense of exact matching. As a consequence, we formulate a weakened version of the classical scattered data interpolation problem, and give a simple and efficient procedure to obtain near-interpolation formulas. Near-interpolants enjoy many remarkable properties, which are very useful from both theoretical and practical points of view (shape preserving properties, operator positivity, subdivision techniques, parallel and multistage computation). Applications of near-interpolants to the representation of surfaces, in particular with faults, are discussed in detail (parameter values, localizing weights, etc.).

§1. Introduction

In many applications, the given function values are subject to errors; hence it is not appropriate to interpolate the function at the data in the sense of exact matching, but it seems more appropriate to approximate the function or, more precisely, to get a relaxed interpolation or near-interpolation. Data requiring near-interpolation by scattered data methods occur in virtually every field of science and engineering. Sources include both experimental results (experiments in chemistry, physics, engineering) and measured values of physical quantities (meteorology, oceanography, optics, geodetics, mining, geology, geography, cartography), as well as computational values (e.g., output from finite element solutions of partial differential equations).

As a consequence of this remark, we formulate a relaxed version of the classical multivariate interpolation problem at scattered data points.

Curve and Surface Fitting: Saint-Malo 1999
Albert Cohen, Christophe Rabut, and Larry L. Schumaker (eds.), pp. 75–84.

ISBN 0-8265-1357-3.

Definition 1. *Given a set of points $S_n = \{x_i,\ i = 1,\dots,n\}$, distinct and generally scattered, in a domain $D \subset \mathbb{R}^s, (s \geq 1)$, with associated values $\{f_i,\ i = 1,\dots,n\}$, and a linear space $\Phi(D)$ spanned by continuous real basis functions $g_j(x;r)$ with $x \in D, r \geq 0$, and $j = 1,\dots,n$, the multivariate near-interpolation problem at scattered data consists in finding a function $F(x;r) \in \Phi(D)$ such that*

$$F(x_i;r) = \sum_{j=1}^{n} a_j g_j(x_i;r) = f_i + \epsilon_i(r), \quad i = 1,\dots,n, \tag{1}$$

and

$$\lim_{r\to 0} \epsilon_i(r) = 0. \tag{2}$$

We observe that r works as a parameter, and the limit case of $F(x;r)$ when r vanishes

$$F(x) \equiv F(x;0) = \lim_{r\to 0} F(x;r)$$

is an interpolation operator. If $F(x;r)$ is specified, then the $\epsilon_i(r)$ in (1) are known; these *near-interpolation errors at the nodes* must not be confused with the unknown errors which affect the corresponding function values f_i. However, it is reasonable to get things so that the $\epsilon_i(r)$ and the errors on f_i are quantities of the same order.

In Section 2 we give a constructive procedure to obtain a wide class of near-interpolation formulas. These enjoy many interesting properties which are listed in Section 3. A crucial point in near-interpolation is the proper choice of the parameter r in (1) and, eventually, of other parameters; the matter is discussed in Section 4. Finally, Section 5 is devoted to the application of near-interpolation to modelling faults.

§2. Construction of Near-Interpolants

To solve the classical interpolation problem, one can consider basis functions which depend on the nodes and, moreover, are cardinal. The method of cardinal basis functions involves selecting continuous *cardinal functions* $g_j : D \to \mathbb{R}$, $(j = 1,\dots,n)$, such that $g_j(x_i) = \delta_{ij}$, $(i = 1,\dots,n)$, where δ_{ij} is the Kronecker delta operator, and setting up the interpolation operator F in the form

$$F(x) = \sum_{j=1}^{n} f_j\ g_j(x).$$

The corresponding near-interpolation problem considers basis functions $g_j(x;r)$, which are no longer cardinal, but $g_j(x;r) \to g_j(x)$ for $r \to 0$. If such $g_j(x;r)$ are given, then

$$F(x;r) = \sum_{j=1}^{n} f_j\ g_j(x;r) = F(x) + \sum_{j=1}^{n} f_j\ [g_j(x;r) - g_j(x)] \tag{3}$$

represents a solution of the near-interpolation problem. In this relation the terms $\epsilon_i(r) = F(x_i;r) - f_i$, $(i = 1,\dots,n)$, are uniquely determined and satisfy (2).

As an example, let us consider the near-interpolant Shepard's formula in the product form

$$\mu(x;r) = \sum_{j=1}^{n} f_j \, \frac{\prod_{k=1,k\neq j}^{n} [d^2(x,x_k)+r]^{\beta}}{\sum_{h=1}^{n} \prod_{k=1,k\neq h}^{n} [d^2(x,x_k)+r]^{\beta}}\,, \tag{4}$$

where $d(x,y)$ is the Euclidean distance between x and y, and $\beta > 0$, or in the equivalent barycentric form

$$\mu(x;r) = \sum_{j=1}^{n} f_j \, \frac{[d^2(x,x_j)+r]^{-\beta}}{\sum_{h=1}^{n} [d^2(x,x_h)+r]^{-\beta}}\,, \quad \mu(x_i;0) = f_i, \quad i = 1,\dots,n. \tag{5}$$

This formula no longer interpolates for $r > 0$, but $\mu(x;r) \to \mu(x)$ as $r \to 0$, where $\mu(x)$ is the well-known Shepard's formula [8, 1].

Examining the structure and the basic idea of Shepard's operator suggests a simple and efficient procedure to obtain an interpolation formula [1]. The corresponding way to obtain a near-interpolation formula is contained in

Definition 2. *Let $\alpha(x,y;r)$, with $x,y \in D$ and $r \geq 0$, be a continuous positive real function such that*

$$\lim_{r\to 0} \alpha(x,y;r) = \alpha(x,y), \tag{6}$$

where $\alpha(x,y) > 0$, if $x \neq y$ and $\alpha(x,y) = 0$, if $x = y$ for all $x,y \in D$. Define now the functions $g_j(x;r)$ by the equations

$$g_j(x;r) = \frac{\prod_{k=1,k\neq j}^{n} \alpha(x,x_k;r)}{\sum_{h=1}^{n} \prod_{k=1,k\neq h}^{n} \alpha(x,x_k;r)}\,, \tag{7}$$

and the near-interpolant $F(x;r)$ by

$$F(x;r) = \sum_{j=1}^{n} f_j \; g_j(x;r) = \sum_{j=1}^{n} f_j \, \frac{\prod_{k=1,k\neq j}^{n} \alpha(x,x_k;r)}{\sum_{h=1}^{n} \prod_{k=1,k\neq h}^{n} \alpha(x,x_k;r)}\,, \tag{8}$$

or equivalently by

$$F(x;r) = \sum_{j=1}^{n} f_j \, \frac{1/\alpha(x,x_j;r)}{\sum_{h=1}^{n} 1/\alpha(x,x_h;r)}\,, \quad F(x_i;0) = f_i, \quad i = 1,\dots,n. \tag{9}$$

Many choices are possible for the function $\alpha(x,y;r)$ in (6); there are no constraints engendered by the set S_n, that is, the distribution of the nodes is irrelevant. Nevertheless, experience suggests identifying α with a radial function

$$\alpha(x,y;r) = \phi(\|x-y\|^2 + r),$$

where $\|\cdot\|$ is a convenient norm. As an example, choosing the Euclidean norm $\|\cdot\|_2$ and

$$\alpha(x,y;r) = (\|x-y\|_2^2 + r)^\beta, \qquad \beta > 0, \tag{10}$$

we obtain from (8) the near-interpolant Shepard's formula (4).

It is often convenient to consider the function $\varphi(t;r) : \mathbb{R}_{\geq 0} \to \mathbb{R}$, associated with $\phi(\|x-y\|^2 + r)$ and defined as $\varphi(t;r) = \phi(t^2 + r)$. We have just seen $\varphi(t;r) = (t^2 + r)^\beta$ in (10); another possible expression that works is $\varphi(t;r) = (t^2+r)^\beta \exp(\gamma\, t^2), (\gamma > 0)$. Considering several functions φ can be useful in order to compare their behaviour and choose the most suitable for use.

§3. Properties of Near-Interpolants

Near-interpolants, as given in Definition 2, enjoy many interesting properties. We list some of them.

A) The near-interpolant $F(x;r)$ in (8) or (9) is a weighted arithmetic mean of the values f_j, $(j = 1,\dots,n)$, since $0 \leq g_j(x;r) \leq 1$ and $\sum_{j=1}^n g_j(x;r) = 1$. As a consequence $F(x;r)$ satisfies the betweeness property $\min_i f_i \leq F(x;r) \leq \max_i f_i$, and reproduces exactly any constant function $f(x) \equiv c$, that is, if $f_i = c$, $(i = 1,\dots,n)$, then $F(x;r) = c$. Moreover, $F(x;r)$, considered as a functional on the set of functions $f : D \to \mathbb{R}$, is linear and positive.

B) If $\alpha(x,y;r)$ is infinitely differentiable with respect to the pth component of $x = (x^{(1)},\dots,x^{(s)})$ for all $x,y \in D$ and $p = 1,\dots,s$, then $F(x;r)$ is also infinitely differentiable with respect to the $x^{(p)}$. For example, choosing $\alpha(x,y;r)$ as in (10), $F(x;r) = \mu(x;r)$ in (4) can be differentiated as many times as desired.

C) If α is a radial function, $F(x;r)$ enjoys some properties of invariance with respect to affine transformations. In particular, with the Euclidean norm, we have that $F(x;r)$ is invariant under translation and rotation, but not scalar invariant.

D) Subdivision techniques can be applied to near-interpolants achieving remarkable results, very well suited for parallel computation [3]. Let us make a partition of the set S_n on the domain D into q subsets S_{n_j}, so that the jth subset, $(j = 1,\dots,q)$, consists of the nodes $x_{j1}, x_{j2},\dots,x_{jn_j}$, with $n_1 + n_2 + \cdots + n_q = n$, and the values $f_{jk_j}, (j = 1,\dots,q; k_j = 1,\dots,n_j)$, correspond to the nodes x_{jk_j}. The indexing of the nodes in the subsets may not depend on the indexing in the set, provided the biunivocity is saved.

Given $S_{n_j} = \{x_{j1}, x_{j2},\dots,x_{jn_j}\}$, $(j = 1,\dots,q)$, let $S_n = S_{n_1} \cup S_{n_2} \cup \cdots \cup S_{n_p}$ and $S_{n_q} \cap S_{n_r} = \emptyset$ for $q \neq r$, then $F(x;r)$ in (9) can be rewritten in the form

$$F_{S_n}(x;r) = \sum_{j=1}^q F_{S_{n_j}}(x;r) \frac{A_j}{\sum_{j=1}^q A_j}, \quad \text{where } A_j = \sum_{k_j=1}^{n_j} 1/\alpha(x, x_{jk_j}; r). \tag{11}$$

E) As a consequence of (11) the following multistage procedure works very well. In the first stage, a given set of nodes $S_{n_1} = \{x_i, i = 1, \dots, n_1\}$ is considered, and the corresponding near-interpolant $F_{S_{n_1}}(x;r)$ is evaluated. In the second stage, it is required to enlarge the considered set S_{n_1}, taking the union of it and another set of nodes $S_{n_2} = \{x_j, j = 1, \dots, n_2\}$. Now the near-interpolating function referred to the union set, i.e., $F_{S_{n_1} \cup S_{n_2}}(x;r)$, with $S_{n_1} \cap S_{n_2} = \emptyset$, can be obtained simply by evaluating the near-interpolant $F_{S_{n_2}}(x;r)$, corresponding to the added set S_{n_2}, and using the relation

$$F_{S_{n_1} \cup S_{n_2}}(x;r) = \frac{F_{S_{n_1}}(x;r)A_1 + F_{S_{n_2}}(x;r)A_2}{A_1 + A_2}, \tag{12}$$

where $A_1 = \sum_{i=1}^{n_1} 1/\alpha(x, x_i; r)$, $A_2 = \sum_{j=1}^{n_2} 1/\alpha(x, x_j; r)$, and A_1 is known. The procedure can be repeated as many times as required.

F) Near-interpolants have the remarkable property that an additional node, say x_{n+1}, can be added to the interpolation set S_n by simply combining an extra term with the original formula. The goal is achieved by using a particular case of (12), that is, the recurrence relation

$$F_{S_{n+1}}(x;r) = \frac{F_{S_n}(x;r)A_n + f_{n+1}\ 1/\alpha(x, x_{n+1}; r)}{A_n + 1/\alpha(x, x_{n+1}; r)},$$

where $A_n = \sum_{k=1}^{n} 1/\alpha(x, x_k; r)$.

G) It is often convenient to extend (8), or better (9), in the following way:

$$F_1(x;r) = \sum_{j=1}^{n} f_j \frac{\tau(x, x_j; \gamma)\ 1/\alpha(x, x_j; r)}{\sum_{h=1}^{n} \tau(x, x_h; \gamma)\ 1/\alpha(x, x_h; r)}, \tag{13}$$

where $\tau(x, y; \gamma)$, with $x, y \in D$ and $\gamma \geq 0$, is a continuous positive real function. Choosing suitably $\tau(x, y; \gamma)$, one can modify the weights in (13) in order either to cancel a useless characteristic, or to introduce a new feature. In particular, it is possible to localize the method considering a factor $\tau(x, y; \gamma)$ rapidly decreasing with distance [2]. The formulas obtained in this way maintain, in general, the analytical and computational properties of the corresponding original ones.

The use of the exponential-type function

$$\tau(x, y; \gamma) = \exp(-\gamma \|x - y\|_2^2) \tag{14}$$

is suggested by McLain [4] for Shepard's formula; he observes that much more accurate results can be obtained in this way. The use of exponential-type weights increases the computational effort, but generally this drawback can be tolerated.

The value of the parameter in the mollifying function $\tau(x, y; \gamma)$ may depend on the nodes, as happens in the popular case [4]

$$\tau(x, x_j; \rho_j) = \left(1 - \frac{\|x - x_j\|_2}{\rho_j}\right)_+^2, \tag{15}$$

where ρ_j is the radius of the circle of support at the point x_j, and $(u)_+ > 0$ if $u > 0$, $(u)_+ = 0$ if $u \leq 0$.

H) The precision of the operator $F(x;r)$ can be increased, considering the Taylor expansion for the function f in each node x_j instead of the function value f_j. This leads to the following extension of (9).

If $f \in C^m(D)$ and $T_j(x)$ is the truncated Taylor expansion for f up to derivatives of order m evaluated at the point x_j and referred to the displacement $h_j = x - x_j$, with $h_j \subset D$, then the operator

$$F_2(x;r) = \sum_{j=1}^{n} T_j(x)\; g_j(x;r), \tag{16}$$

near-interpolates to $T_j(x)$ at $x = x_j$. In this form, $F_2(x;r)$ reproduces exactly algebraic polynomials of degree $\leq m$.

Combining the modifications in (13) and (16), we have

$$F_3(x;r) = \sum_{j=1}^{n} T_j(x)\; \frac{\tau(x,x_j;\gamma)\; 1/\alpha(x,x_j;r)}{\sum_{h=1}^{n} \tau(x,x_h;\gamma)\; 1/\alpha(x,x_h;r)}\,. \tag{17}$$

Obviously, the technique calls for additional derivative values that are not normally available as data. A more practical solution is discussed below.

K) For simplicity, we refer here to an Euclidean radial function $\alpha(x,y;r) = \phi(\|x-y\|_2^2 + r)$, because in this case the procedure is well established. The primary modifications required involve using $\tau(x,y;\gamma)$ to localize the overall approximation, and replacing f_j with a suitable "local approximation" to the surface. To carry out the approximation (17), a practical way is to get, in a first stage, local approximants $M_j(x)$ to $f(x)$ at the points $x_j, (j = 1,\ldots,n)$, obtained by means of the moving weighted least-squares method using weight functions with reduced compact support. Then, in a second stage, the near-interpolating operator is expressed as a convex combination of the local approximants

$$F_4(x;r) = \sum_{j=1}^{n} M_j(x)\; \frac{\tau(x,x_j;\gamma)\; 1/\phi(\|x-x_j\|_2^2 + r)}{\sum_{h=1}^{n} \tau(x,x_h;\gamma)\; 1/\phi(\|x-x_h\|_2^2 + r)}\,. \tag{18}$$

In particular, by (18), (10), and (14),

$$\mu_1(x;r) = \sum_{j=1}^{n} M_j(x)\; \frac{\exp(-\gamma\|x-x_j\|_2^2)\; (\|x-x_j\|_2^2 + r)^{-\beta}}{\sum_{h=1}^{n} \exp(-\gamma\|x-x_h\|_2^2)\; (\|x-x_h\|_2^2 + r)^{-\beta}}\,, \tag{19}$$

which extends (5).

Very good performance is achieved by a version of (18) which uses quadratic approximations for $M_j(x)$, and mollifying functions given by (15). This method has been developed by Franke and Nielson [4], and Renka [7] for Shepard's operator.

§4. Determining Parameter Values

The near-interpolating operator (19) works very well in a large variety of cases. Our attention is here focused on finding values of the parameters r, β and γ, which can be regarded as "optimal" from a practical viewpoint. The considerations which follow are mainly based on experiments.

A relatively small increase of the parameter r in (19) in some right neighbourhood of zero has a considerable effect on the behaviour of $\mu_1(x;r)$. In fact, if r is small, x fixed and near to the node x_{j^*}, then the value of $g_{j^*}(x;r)$ equals nearly one; but, if r increases, $g_{j^*}(x;r)$ decreases. Since $\sum_{j=1}^{n} g_j(x;r) = 1$, diminishing of $g_{j^*}(x;r)$ makes the other weight values $g_j(x;r)$, $j \neq j^*$, increase. Summing up, if the weight attributed to f_{j^*} in $\mu_1(x;r)$ decreases, then $\mu_1(x_{j^*};r) = f_{j^*} + \epsilon_{j^*}(r)$ diverges from f_{j^*}, namely $\epsilon_{j^*}(r)$ increases and reduces the accuracy of $\mu_1(x_{j^*};r)$.

Introducing the parameter r in (19), and in particular in (5), has the effect that, in general, the gradient of the rendered surface is not zero at the nodes. As a consequence, the surface is considerably smoother than for $r = 0$. However, if r is too small, the first derivatives of $\mu_1(x;r)$ are highly oscillating and their values are nearly zero. Clearly, the goal is to choose an "optimal" value of r, such that $\mu_1(x;r)$ does not exhibit the characteristic irregularities of the basic Shepard's formula, but at the same time, it maintains a sufficient computational accuracy, in particular at the nodes.

The search for the optimal value of r can be done by many applications of (19) with different values of the parameter, and then by choosing that value which minimizes the global root mean square error. Although this is currently considered in the literature, the estimate of r is not a simple matter; in a sense, it can be compared with the analogous difficult problem of computing the optimal value of the parameter in multiquadric interpolants.

The optimal value of the parameter β has been determined with particular attention to computational accuracy. The performance analysis on some test functions proposed by Franke leads to prefer the value $\beta = 3/2$.

As for the optimal value of the parameter γ in the strongly localizing function (14), McLain has proposed $\gamma = 1.62n/\mathrm{diam}(D)$, where n is the number of nodes and $\mathrm{diam}(D)$ is the diameter of D. However, this value is, in general, too large, whereas it is sufficient to consider for γ a value of the order of tens.

§5. Application to Modelling Faults

Using the near-interpolating operator $\mu_1(x;r)$ of (19), with a suitable value of the parameter r, instead of the corresponding interpolating operator $\mu_1(x;0)$, increases considerably the performance of the approximation in a rich variety of applications, because it permits consideration of supplementary information connected with the characteristics of the examined problem. A typical case occurs with surface discontinuities, in particular faults, which are frequently met when modelling geological surfaces.

Following Shepard [8], we observe that, if some physical barrier such as a fault separates the set of nodes, the relationship between nodes on the opposite sides of the barrier may be attenuated. Through the inclusion of barriers, a user may specify discontinuities in the metric space in which the distance between two points is calculated to simulate this attenuation. Suppose a "detour" of length $b(x, x_k)$ were required to go over the barrier between x, the current near-interpolation point, and the node x_k. The quantity $b(x, x_k)$ is considered the strength of the barrier, and an effective distance between x and x_k is given by

$$d^*(x, x_k) = \sqrt{[d(x, x_k)]^2 + [b(x, x_k)]^2}.$$

This definition is general so that if no barrier separates x and x_k, then $b(x, x_k) = 0$ and $d^*(x, x_k) = d(x, x_k)$. Because of the discontinuity in effective distance as the near-interpolation point x crosses the barrier, the rendered surface will be discontinuous at the barrier.

Since extensive tests [4] have shown that the modified quadratic Shepard's method performs very well for a variety of data sets, Franke and Nielson [5] have chosen it as a basis to investigate the problem of simulating faults. Our approach uses instead the near-interpolant (19), with significant differences in distance penalty, localizing functions, fault forms, etc., as compared to Franke and Nielson.

The possibility of having to model faults can occur in different ways [5]; to save space, we limit our attention to the following case: there is a known fault line $\Gamma \subset D \subset \mathbb{R}^2$, in a known location, with a known jump. More complicated situations (see, e.g., [5,6]) require extensive considerations that will be discussed in a further work.

As a first step, it is convenient to focus on the basic situation in which the fault line Γ is a segment l and, moreover, the jump is constant along l. Then, a known polygonal curve can be considered as a fault line; in fact, the reduction to the case of a fault line segment is straightforward by using the subdivision procedure considered in Section 3. In principle, any curve can be considered as a fault line, provided it is well approximated by a polygonal. Another extension consists in considering a jump varying along the fault line. Also the reduction to the basic case is now possible, subdividing the fault line into a convenient number of segments, and using a mean value of the jump for each segment.

To deal with the basic case, we modify the value of the parameter r in (19) in order to take the jump into account. Let x be the near-interpolation point, x_k a node and l^* the segment joining x and x_k. Then for $x, x_k \notin l$ we set

$$r = \begin{cases} r_{opt}\,, & \text{if} \quad l \cap l^* = \emptyset, \\ b(x, x_k), & \text{if} \quad l \cap l^* \neq \emptyset, \end{cases}$$

where the quantity r_{opt} is the optimal value obtained for r in (19) on the opposite sides of the fault and $b(x, x_k)$ represents the "effort" required to go over the barrier, due to the discontinuity dividing the two points. If the jump

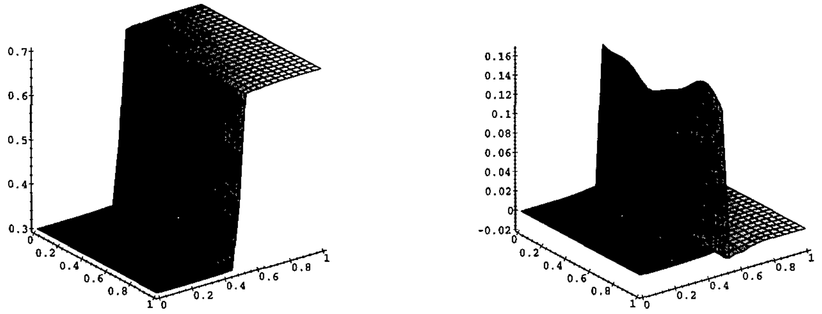

Fig. 1. Function $f_1(x,y)$: near-interpolation and signed error surfaces.

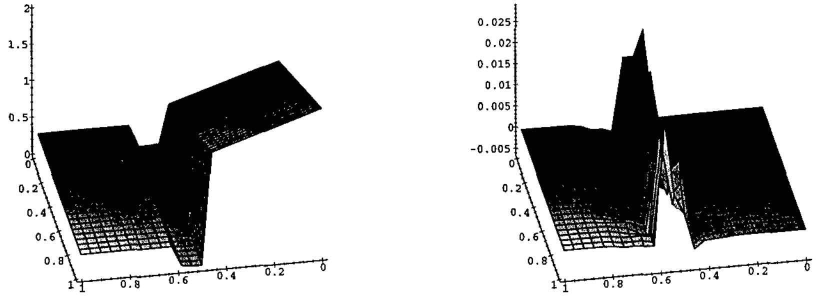

Fig. 2. Function $f_2(x,y)$: near-interpolation and signed error surfaces.

is constant or almost constant along l, it is possible to simply set $b(x, x_k) = h$, where h is the jump size.

Formula (19), after these adjustments in the parameter r, gives results quite good both for the appearance of the graphic representation and the accuracy in computation. Comparing the rendered surface with the one obtained by the modified quadratic Shepard's formula shows that the introduction of the parameter r gives a smoother surface which is closer to the approximated function.

Our procedure has been used to fit the test function proposed by Franke and Nielson [5] using their set of nodes. Numerous tests were also made on other surfaces. We present two examples of the rendered surfaces and the signed error surfaces for the functions

$$f_1(x,y) = \begin{cases} 0.3, & \text{if } 0 \le x < 0.5, \\ 0.7, & \text{if } x \ge 0.5; \end{cases} \qquad f_2(x,y) = \begin{cases} y - x + 1, & \text{if } 0 \le x < 0.5, \\ 0, & \text{if } 0.5 \le x < 0.6, \\ 0.3, & \text{if } x \ge 0.6, \end{cases}$$

defined on the unit square (see Fig. 1 and Fig. 2). We used the parameter values $r = 0.0036, \beta = 1.5, \gamma = 24$, and $r = 0.0025, \beta = 1.5, \gamma = 30$ respectively, and once again the set of nodes of Franke and Nielson. The errors can be considerably reduced by adding more information on the faults; in fact, the employed set of nodes is not obviously an ad hoc choice.

Acknowledgments. This research was supported by the Italian Ministry of Scientific and Technological Research and the University of Turin within the project "Advanced Numerical Methods for Scientific Computing".

References

1. Allasia, G., A class of interpolating positive linear operators: theoretical and computational aspects, in *Recent Developments in Approximation Theory, Wavelets and Applications*, S. P. Singh (ed.), Kluwer, Boston, 1995, 1–36.
2. Allasia, G., and R. Besenghi, Properties of interpolating means with exponential-type weights, in *Curves and Surfaces in Geometric Design*, P.-J. Laurent, A. Le Méhauté, and L. L. Schumaker (eds.), A. K. Peters, Wellesley MA, 1994, 1–8.
3. Allasia, G., and P. Giolito, Fast evaluation of cardinal radial basis interpolants, in *Surface Fitting and Multiresolution Methods*, A. Le Méhauté, C. Rabut, and L. L. Schumaker (eds.), Vanderbilt University Press, Nashville, 1997, 1–8.
4. Franke, R., Scattered data interpolation: tests of some methods, Math. Comp. **38** (1982), 181–200.
5. Franke, R., and G. M. Nielson, Surfaces approximation with imposed condition, in *Surfaces in CAGD*, R. E. Barnhill and W. Boehm (eds.), North-Holland, Amsterdam, 1983, 135–146.
6. Parra, M. C., M. C. López de Silanes and J. J. Torrens, Vertical fault detection from scattered data, J. Comp. Appl. Math. **73** (1996), 225–239.
7. Renka, R. J., Multivariate interpolation of large sets of scattered data, ACM Trans. Math. Softw. **14** (2) (1988), 139–148.
8. Shepard, D., A two-dimensional interpolation function for irregularly spaced data, *Proc. 23rd Nat. Conf. ACM*, 1968, 517–524.

Renata Besenghi, Giampietro Allasia
Dipartimento di Matematica
Università di Torino
I-10123 Torino, Italy
besenghi@dm.unito.it, allasia@dm.unito.it